

СЕКЦІЯ 10

АВТОМАТИЗАЦІЯ ЕКСПЕРИМЕНТАЛЬНИХ ДОСЛІДЖЕНЬ

УДК 621.3.087.44

РОЗПОДІЛЕНА СИСТЕМА ЗБОРУ ДАНИХ НА БАЗІ МЕРЕЖЕВИХ ІНТЕЛЕКТУАЛЬНИХ СЕНСОРІВ

Богомазов С. А.

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

E-mail: sbogmzv@gmail.com

Суттєвою перешкодою в розвитку провідникових та безпроводникових мереж інтелектуальних датчиків є відсутність єдиного мережевого стандарту. Зараз використовуються десятки типів різноманітних інтерфейсів (RS-485, HART, USB, 4-20mA, IEEE-488) та промислових мереж (Profibus, Fieldbus, DeviceNet, Interbus, CANbus, Modbus, LIN). Перед виробниками датчиків постає завдання вибору типу цифрового інтерфейсу та комунікаційного протоколу, оскільки виробництво однотипних інтелектуальних датчиків для кожної з популярних в теперішній час мереж є економічно не вигідним.

Стандарти групи IEEE 21451 (раніше відомі як IEEE 1451) уніфікують інтерфейс між інтелектуальним датчиком та мережею [1]. В стандартах IEEE 21451 описані два класи пристроїв: TIM (Transducer Interface Module) та мережевий процесор NCAP (Network Capable Application Processor). Окрім того, визначається концепція TEDS (Transducer Electronic Data Sheet) – електронна специфікація датчика, що забезпечує самоідентифікацію пристрою в мережі. Специфікація TEDS містить такі дані, як код моделі, серійний номер, дату випуску та калібровочні дані, одиниці виміру, дату калібровки тощо [2].

Meta TEDS детально описує наявні та доступні у TIM канали, які більш детально описані в Channel TEDS кожного з каналів. У Channel TEDS описується інформація, що відноситься до фізичного рівня. Калібровочна інформація для кожного каналу записується у електронну документацію – Calibration TEDS [3].

На основі мікроконтролера MSP430G2553 (Texas Instruments) з ультранизьким енергоспоживанням та температурного датчика DS18B20 було реалізовано модуль TIM демонстраційного інтелектуального сенсора температури. Програмно-апаратне забезпечення модуля NCAP реалізовано на основі Java-мікросервера, створеного на базі одноплатного комп'ютера Cubieboard із операційною системою Linux. Програмне забезпечення модуля TIM реалізовано на мові embedded C, програмне забезпечення модуля NCAP реалізовано на мові програмування Java.

Для мікроконтролерного пристрою MSP430G2 розроблено програмне забезпечення, що реалізує функції модуля TIM у відповідності із стандартами IEEE 21451. Протокол передачі даних ТІІ (Transducer Independent Interface) між модулем NCAP і модулем TIM не залежить від фізичного рівня. Процес

передачі ініціює модуль NCAP, який звертається з командою до конкретного ТІМ. Коли ТІМ отримує і розпізнає команду, ця команда обробляється і надсилається відповідь до модуля NCAP. Команди, які надходять до ТІМ від NCAP, мають довжину три байти. Після надходження команди ТІМ надсилає 4-х байтову мета-відповідь. У розробленому модулі підтримується чотири команди стандарту IEEE 21451: `readAvailableChannels()` – прочитати інформацію про всі доступні канали в межах одного ТІМ; `readMetaTeds()` – прочитати Meta TEDS; `readChannelTeds()` – прочитати Channel TEDS; `singleMeasurement()` – виконати вимірювання.

Розроблено бібліотеку підтримки функцій модуля NCAP за стандартом IEEE 21451 для Java-машини на основі одноплатного комп'ютера Cubieboard, яка складається з наступних пакетів: `*.ncap.channels` – містить класи для роботи з каналами ТІМ; `*.ncap.teds` – містить класи для роботи з TEDS; `*.ncap.tii` – містить класи для організації передачі інформації; `*.ncap.util` – містить допоміжні класи для роботи з бібліотекою; `*.ncap` – містить основний клас для роботи з бібліотекою.

Клас `Msp430Tii` реалізує функції Java-інтерфейсу ТІІ для обміну інформацією з модулем ТІМ. Оскільки реалізація передбачає передачу даних через інтерфейс UART, то даний клас використовує бібліотеку JSSC. В класі `Msp430ChannelsManager` реалізовано функцію отримання інформації про всі доступні канали в ТІМ. Дана функція надсилає до ТІМ команду – прочитати всі доступні канали. ТІМ надсилає до NCAP відповідь у вигляді байтового масиву. У байтовому масиві містяться назви та `id` доступних каналів. Клас `Msp430FunctionManager` підтримує функцію зчитування Meta TEDS, функцію зчитування Channel TEDS, функцію отримання даних від датчиків. Клієнтська частина розподіленої системи збору даних реалізована на основі RESTful архітектури [4].

Реалізація розподіленої системи збору даних на основі стандарту IEEE 21451 дозволила спростити підключення перетворювачів до вимірювальних приладів та мереж шляхом реалізації єдиних інтерфейсів та механізмів самоконфігурування. Використання платформи-незалежних Java-технологій забезпечило незалежність програмних рішень від апаратної реалізації мережевого процесора та дозволило використовувати запропоновані рішення для широкого спектру операційних систем та апаратного забезпечення.

Ключові слова: інтелектуальний сенсор, стандарт IEEE 21451, збір даних.

Література

- [1] IEEE Standards Association. (2007). *IEEE Standard for a Smart Transducer Interface for Sensors and Actuators*. doi:10.1109/IEEESTD.2007.4346346.
- [2] A. Kumar, V. Srivastava, M. K. Singh, and G. P. Hancke, Current status of the IEEE 1451 standard based sensor applications, *IEEE Sensors J.*, vol. 15, no. 5, pp. 2505-2513, May 2015.
- [3] V. Viegas, O. Postolache, and J. M. Dias Pereira, Transducer Electronic Data Sheets: Anywhere, Anytime, Anyway, *Electronics* №8 (11), p.1345, 2019. doi: 10.3390/electronics8111345.
- [4] К. В. Береговий, та С. А. Богомазов, “Реалізація мережі інтелектуальних сенсорів на базі вбудованих Java-модулів”, на *XVIII н.-т. к. студ. та мол. уч. Гіротехнології, навігація, керування рухом та конструювання авіаційно-космічної техніки*, Київ, 2015, с.10.